

고성능 HEVC 복호기를 위한 효과적인 Sample Adaptive Offset 하드웨어 설계

박승용*, 이동원**, 류광기*

*한밭대학교 정보통신전문대학원, **전남대학교 기계시스템공학부
e-mail: srrr.kr@gmail.com, mems@jnu.ac.kr, kkryoo@hanbat.ac.kr

The Hardware Design of Effective Sample Adaptive Offset for High Performance HEVC Decoder

Seungyong Park*, Dongweon Lee**, and Kwangki Ryoo*

*Graduate School of Information and Communication, Hanbat National University

**School of Mechanical System Engineering, Chonnam National University

요 약

본 논문에서는 고성능 HEVC(High Efficiency Video Coding) 복호기 설계를 위한 효율적인 SAO(Sample Adaptive Offset)의 하드웨어 구조 설계에 대해 기술한다. SAO는 양자화 등의 손실 압축에 의해 발생하는 정보의 손실을 보상하는 기술이다. 하지만 HEVC의 최대 블록 크기인 64x64 단위를 화소 단위 연산을 수행하기 때문에 높은 연산시간 및 연산량이 요구된다. 따라서 본 논문에서 제안하는 SAO 하드웨어 구조는 8x8 단위를 처리하는 연산기로 구성하여 하드웨어 면적을 최소화하였고, 내부 레지스터를 이용하여 64x64 블록 크기를 지원한다. 또한 기존 SAO의 top-down 블록분할 구조에서 down-top 블록분할 구조로 설계하여 연산시간 및 연산량을 최소화 하였다. 제안하는 하드웨어 구조는 Verilog HDL로 설계하였으며, TSMC 칩 공정 0.18 μ m 셀 라이브러리로 합성한 결과 동작 주파수는 250MHz, 전체 게이트 수는 45.4k 이다.

1. 서론

최근 멀티미디어 서비스는 HD(High Definition) 및 UHD(Ultra High Definition) 등과 같은 고해상도/고화질 영상을 요구하고 있다. 그러나 현재 가장 우수한 압축 효율을 가진 H.264/AVC 표준과 같은 최신 압축 표준으로도 고해상도/고화질 영상의 데이터를 저장하거나 현재의 통신채널로 전송하기에는 상당한 비용이 발생한다. 이러한 문제에 대응하고자, ISO/IEC MPEG(Moving Picture Experts Group)과 ITU-T VCEG(Video Coding Experts Group)은 2010년 4월 독일 Dresden에서 개최된 제1회 JCT-VC (Joint Collaborative Team on Video Coding) 표준화 회의를 시작으로 차세대 영상 압축 표준인 HEVC(High Efficiency Video Coding) 표준 제정 프로젝트를 시작하였다. 현재 표준화가 진행되고 있는 HEVC 표준은 기존의 H.264/AVC 표준과 같은 움직임 보상 변환부호화 압축 기술에 기반을 하며, 보다 향상된 다양한 요소 기술을 적용함으로써 압축 효율을 극대화하고 있다[1].

H.264/AVC 표준은 루프내 필터링 기술에서 영상의 블록화 현상을 제거하기 위한 디블록킹 필터만을 포함하였지만, HEVC에서는 디블록킹 필터, 양자화 등의 손실 압축에 의해 발생하는 정보의 손실을 보상하기 위한 SAO(Sample Adaptive Offset) 및 ALF(Adaptive Loop Filter)와 같은 새로운 기술을 추가적으로 포함되어 주관

적 화질 뿐만 아니라 압축 효율을 향상시키는 구조를 가지고 있다. HEVC의 루프내 필터링 기술에서 SAO(Sample Adaptive Offset) 기술은 양자화 등의 부호화 과정을 통해 발생하는 원본 영상과 복원 영상 간의 왜곡을 화소(Sample) 단위의 offset을 통해 보상하여, 주관적 화질 및 부호화 효율을 같이 향상시키는 것이다. HEVC에서 처리하는 최대 블록 크기는 64x64로 화소 단위 연산을 수행하는 SAO에서는 높은 연산량과 연산시간을 차지한다[2].

본 논문에서는 SAO의 높은 연산량과 연산시간, 하드웨어 설계시 면적을 최소화할 수 있는 하드웨어 구조를 제안한다. 제안하는 하드웨어 구조는 8x8 연산기로 구성되며, 내부 레지스터를 이용하여 64x64 연산에 필요한 화소를 저장하는 구조로 구현하였다. 또한, 기존 SAO의 블록분할 구조는 top-down으로 최적의 SAO 블록분할과 SAO type을 결정하기 위해 0-depth부터 3-depth까지 4번을 반복하여 수행한다. 하지만 본 논문에서 제안하는 블록분할 구조는 down-top 방식으로 최하위 depth를 처음 연산하여 상위 depth에서 하위 depth의 연산결과를 사용하는 구조로 설계하였다. 그 결과 제안하는 down-top 방식은 3-depth부터 0-depth까지 1번만 수행하여 연산시간 및 하드웨어 면적을 최소화 하였다.

본 논문의 구성은 다음과 같다. 2장에서는 기존 SAO의 기법을 기술하며, 3장에서는 제안하는 SAO 하드웨어 구조를 기술한다. 4장에서는 하드웨어 합성 결과를 기술하며,

마지막으로 5장에서는 결론으로 끝을 맺는다.

2. 기존 SAO 기법

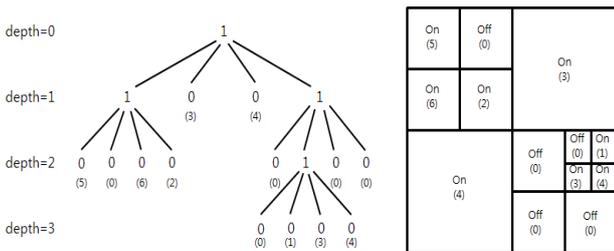
SAO(Sample Adaptive Offset) 기법은 양자화 등의 부호화 과정을 통해 발생하는 원본 영상과 복원 영상간의 왜곡을 화소(Sample) 단위의 offset을 통해 보상함으로써 주관적 화질 및 부호화 효율을 향상시키는 것이다.

식(1)과 같이, 부호화기가 N 개의 화소에 대해 원본 화소(pic_n)와 복원 화소(rec_n) 간의 오차를 평균적으로 최소화하는 offset을 계산하고, 복호기에 전송하여 식(2)와 같이 부/복호화기가 동일한 방법으로 복원 화소의 왜곡을 보상할 수 있다. 특히 왜곡의 정도가 다른 화소를 구분해서 서로 다른 offset을 적용하는 적응적인(Adaptive) offset 보상 방법을 사용함으로써, 영상의 특성에 따라 부분적으로 다르게 발생하는 왜곡을 효율적으로 최소화할 수 있다 [3].

$$offset = \frac{\sum_{n=0}^{N-1} (\text{pic}_n - \text{rec}_n)}{N} \quad (1)$$

$$\text{rec}'_n = \text{rec}_n + offset \text{ where } n=0,1,\dots,N-1 \quad (2)$$

그림 1과 같이 quadtree 기반의 CU 분할 방법과 동일한 방식으로 LCU(Largest CU) 블록을 최대 4-depth까지의 하위 블록으로 분할하여, 각 하위 블록에는 서로 다른 SAO type이 적용될 수 있다. 부호화기는 율-왜곡 최적화 방법을 사용하여 부호화 효율 측면에서 최적인 SAO 블록 분할 여부 및 SAO type을 결정하고, 이에 해당하는 정보를 복호화기에 전송한다.

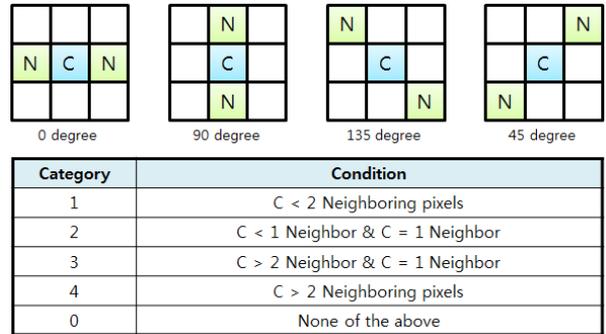


(그림 1) SAO 블록분할 구조 및 SAO type 적용 예

<표 1> SAO type 정보

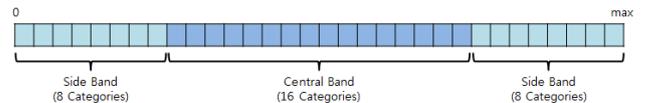
sao_type_idx	SAO type to used		Number of categories
0	None		0
1	Edge Offset	0-degree	4
2		90-degree	4
3		135-degree	4
4		45-degree	4
5	Band Offset	Central bands	16
6		Side bands	16

분할된 각 하위 블록에 적용 가능한 SAO type은 표 1과 같다. SAO를 적용하지 않는 방법, EO(Edge Offset) 및 BO(Band Offset)으로 구분할 수 있다. EO 및 BO는 해당 화소의 카테고리를 구분하기 위한 방법에 따라 EO는 0-degree, 90-degree, 135-degree, 45-degree 총 4가지 방향으로 구분되며, BO는 central band와 side band로 구분된다.



(그림 2) EO(Edge Offset)의 카테고리

그림 2는 EO의 카테고리를 나타낸다. 현재 화소와 4개의 방향에 위치하는 주변 화소사이의 차이를 비교하여 카테고리를 정하는 방식이다.



(그림 3) BO(Band Offset)의 카테고리

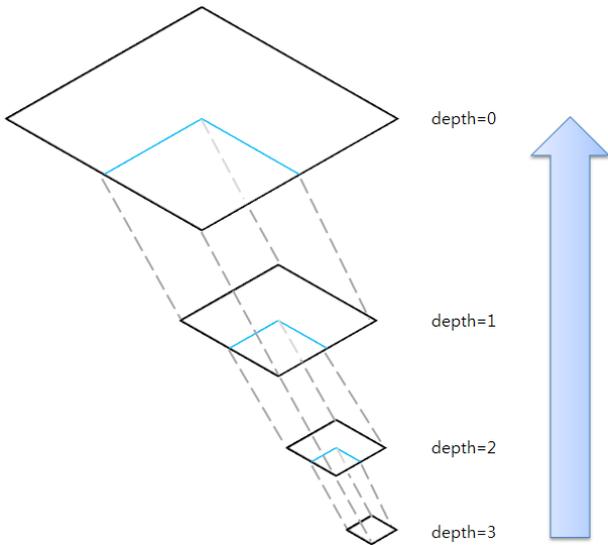
그림 3은 BO의 카테고리를 나타낸다. 화소가 속할 수 있는 값을 일정한 크기의 32개 구간으로 구분하고, 각 구간 단위로 식 1과 같은 방법을 통해 각각의 offset이 결정된다. BO는 16개의 카테고리를 갖는 2개의 SAO type으로 구분되며, BO가 적용되면 16개의 카테고리를 위해 16개의 offset이 결정되고 전송된다.

SAO 기법은 율-왜곡 최적화 방법을 사용하여 부호화 효율 측면에서 최적인 SAO 블록분할 여부 및 SAO type을 결정한다. top-down 방식으로 0-depth의 율-왜곡 비용과 1-depth의 율-왜곡 비용과 비교하여, 최적의 율-왜곡 성능을 보이는 SAO 블록분할 구조 및 SAO type을 결정한다. 1-depth이 결정되면 위와 같은 방법으로 1-depth와 2-depth의 율-왜곡 비용을 비교하게 된다. 총 3-depth까지 수행을 하며 최적의 SAO 블록분할과 SAO type을 결정한다.

이와 같은 수행 방법은 0-depth에서 3-depth까지 4번을 수행해야 하며 하드웨어로 구현시 각 depth에 해당하는 연산기를 설계해야 하는 문제점이 발생한다. 또한, HEVC 부호기 및 복호기의 성능에도 큰 영향을 미치는 결과를 가진다.

3. 제안하는 SAO 하드웨어 구조

본 논문에서 제안하는 SAO 하드웨어 구조는 더블클링 필터에서 처리되는 영상의 블록크기를 고려하여 4x4 블록 크기의 연산기 4개를 사용하여 8x8 블록 크기의 연산기 구조로 설계하였다. 또한, 기존 SAO 기법의 top-down 방식에서 down-top 방식으로 설계하여 SAO 블록분할과 SAO type을 결정하는데 수행되는 연산시간을 최소화 하였다. 그림 4는 SAO 블록분할과 SAO type을 결정하는 down-top 방식을 나타낸다.



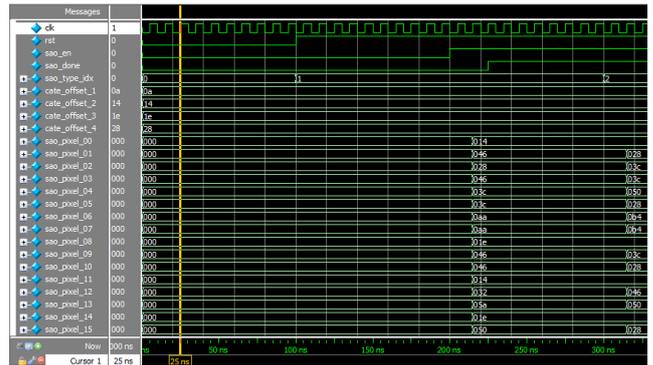
(그림 4) 제안하는 down-top 방식

제안하는 down-top 방식은 최하위 depth인 3-depth부터 SAO 연산을 수행하며, 3-depth의 SAO 카테고리 결과들을 2-depth에서 사용하는 방식을 가진다. 또한, 3-depth와 2-depth의 읍-왜곡 비용을 비교하여 최적의 SAO type과 SAO 블록분할을 결정한다. 상위 depth 연산에 필요한 화소는 내부 레지스터에 저장하여 블록의 경계 부분만 SAO를 수행하는 구조로 설계하였다.

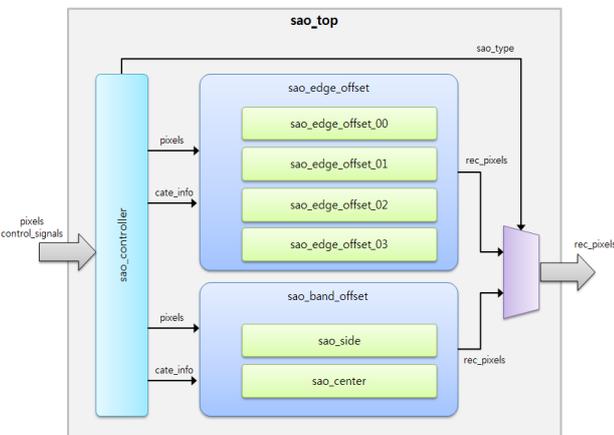
그림 5는 제안하는 SAO 하드웨어 구조이며, sao_top은 sao_controller, sao_edge_offset, sao_band_offset 총 3개의 모듈로 구성된다. sao_controller 모듈은 SAO를 제어하는 기능을 담당하며, sao_edge_offset 모듈은 EO으로 현재 화소와 주변 화소들의 차이로 offset 값을 적용한다. sao_edge_offset 모듈의 현재 화소와 주변 화소들의 차이를 구분하는 방식은 LUT(Look Up Table)를 사용하여 1 사이클에 동작할 수 있도록 구현하였다.

sao_band_offset 모듈은 BO으로 화소를 0부터 255까지 32구간으로 나누어 각 offset 값을 적용한다. 최종으로 offset이 적용된 화소를 부호기에서 전송된 SAO type으로 선택하여 출력한다.

본 논문에서는 HEVC 표준 참조 소프트웨어 HM-8.0에서 비트스트림을 추출하였고, 제안한 하드웨어 구조를 통해 시뮬레이션 한 결과 참조 소프트웨어에서 추출한 비트스트림과 비교하여 정상적으로 동작함을 확인하였다[4]. 그림 6은 SAO offset 연산 시뮬레이션 결과 파형이며, 1 사이클 동안 4x4 블록크기에 해당하는 16개 화소를 출력한다. 총 64x64 블록크기에 해당하는 화소를 출력하기 위해 128 사이클이 소요된다.



(그림 6) SAO offset 연산 시뮬레이션 결과 파형



(그림 5) 제안하는 SAO 하드웨어 구조

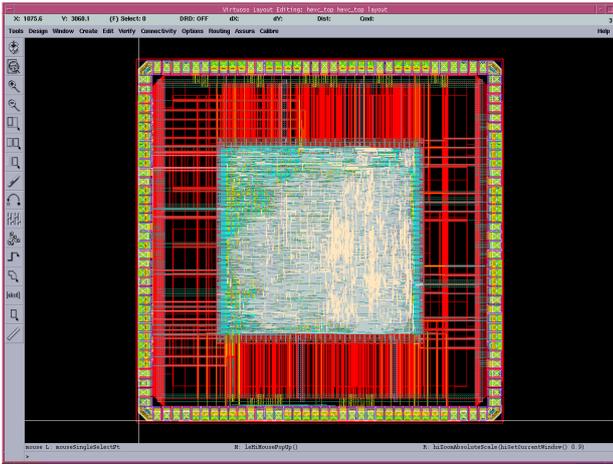
4. 하드웨어 합성 결과

본 논문에서 제안하는 SAO 하드웨어 구조는 Verilog HDL로 설계 하였다. 하드웨어 합성은 IDEC에서 지원하는 CAD tool을 사용하였으며, TSMC 0.18μm 셀 라이브러리로 합성하였다. 표 2는 제안하는 SAO 하드웨어 구조의 합성결과를 나타낸다.

<표 2> 제안하는 SAO 하드웨어 합성 결과

구분	결과
지원 해상도	1080p
공정	TSMC 0.18μm
동작 주파수	250MHz
LCU 당 인코딩 사이클 수	128
게이트 (k)	45.4k

그림 7은 Virtuoso Layout Editing 칩 설계 결과 화면이다.



(그림 7) Virtuoso Layout Editing 칩 설계 결과 화면

5. 결론

본 논문에서는 루프내 필터의 SAO를 위한 연산시간과 연산량, 하드웨어 면적을 최소화하기 위해 8x8 블록 연산기를 사용하였고, LCU인 64x64을 지원하기 위해 내부 레지스터를 사용하여 8x8 블록 경계의 화소들을 저장하는 방식으로 구현하였다. 또한 기존의 SAO 블록분할과 SAO type을 결정하기 위한 top-down 방식을 down-top 방식으로 설계하여 불필요한 연산 시간을 최소화하였다.

제안한 하드웨어 구조는 TSMC 0.18 μ m 셀 라이브러리로 합성한 결과 최대 동작 주파수는 250MHz이고, 게이트 수는 45.4k이다. 또한, 하나의 LCU를 부호화 하는데 128 사이클이 소요된다.

감사의 글

본 논문은 교육과학부와 한국연구재단의 지역혁신인력 양성사업 및 지식경제부 출연금으로 수행한 ETRI SW-SoC 융합 R&BD 센터와의 공동 연구의 결과입니다.

참고문헌

- [1] JCT-VC of ISO/IEC MPEG and ITU-T VCEG, "Meeting report of the first meeting of the JCT-VC," JCT-VC doc. JCTVC-A200, April, 2010.
- [2] B. Bross, W.-J. Han, G. J. Sullivan, J.-R. Ohm, and T. Wiegand, "High efficiency video coding (HEVC) text specification draft 8," Document of Joint Collaborative Team on Video Coding, JCTVC-J1003,

July, 2012.

- [3] JCT-VC of ISO/IEC MPEG and ITU-T VCEG, "CE13: Sample Adaptive Offset with LCU Independent Decoding," JCT-VC doc. JCTVC-E049, January, 2010.
- [4] "HM8: High efficiency video coding HEVC test model 8." [Online]. Available: https://hevc.hhi.fraunhofer.de/svn/svn_HEVCSoftware/tags/HM-8.0/
- [5] W.-S. Kim, "AHG6: SAO parameter estimation using non-deblocked pixels," Joint Collaborative Team on Video Coding, JCTVC-J0139, July, 2012.
- [6] Y.-W. Huang, E. Alshina, I. S. Chong, W. Wan, and M. Zhou, "Description of Core Experiment 1 (CE1): Sample adaptive offset filtering," Joint Collaborative Team on Video Coding, JCTVC-H1101, February, 2012.
- [7] C.-M. Fu, C.-Y. Chen, Y.-W. Huang, S. Lei (MediaTek), S. Park, B. Jeon (LGE), A. Alshin, E. Alshina (Samsung) "Sample Adaptive Offset for Chroma" Document of Joint Collaborative Team on Video Coding, JCTVC-F057, July, 2011.
- [8] Joint Video Team of ITU-T VCEG and ISO/IEC MPEG, ITU-T Rec. H.264 | ISO/IEC 14496-10 AVC, 2005.